

Menggunakan struktur data

1. Struktur data apa yang anda pilih? Mengapa anda memutuskan menggunakan struktur data tersebut?
2. Operasi apa saja yang akan/sudah anda buat terhadap struktur data tersebut?
3. Jelaskan apa yang telah anda lakukan terhadap cara akses data agar algoritma anda berjalan secara efisien dan tunjukkan bagian program anda yang mana yang mendukung hal tersebut

Mengimplementasikan user interface

4. Simulasi aplikasi apa yang sudah anda buat? Mengacu ke sub-modul yang mana dari soal?

Melakukan instalasi software tools pemrograman

5. Apa pertimbangan anda dalam memilih tools Bahasa pemrograman yang sesuai dengan kebutuhan dan lingkungan pengembangan?

Melakukan kode dengan prinsip sesuai guidelines dan best practices

6. Guidelines mana pada penulisan kode yang sudah anda ikuti? Tunjukkan buktinya pada salah satu bagian dari program anda!
7. Tunjukkan dan jelaskan bagian dari program anda yang telah mengimplementasikan kemudahan interaksi sesuai dengan standar yang berlaku

Menggunakan library atau komponen pre-existing

8. Jelaskan bagaimana mengidentifikasi class unit-unit reuse (dari aplikasi lain) yang sesuai
9. Tunjukkan di program anda bagian mana yang memiliki ketergantungan antar unit/modul/class

Membuat dokumen kode program

10. Bagaimana cara anda mengetahui apakah tools untuk generate dokumentasi program telah menghasilkan dokumen kode program sesuai dengan spesifikasi yang diminta

Melakukan debugging

11. Bagaimana cara anda menelusuri galat/error yang terjadi pada program anda? Berikan contoh sebuah galat pada program anda dan cara anda memperbaiki

Menggunakan Struktur Data

1. Struktur data apa yang anda pilih? Mengapa?

Struktur data yang dipilih adalah **tabel relasional dalam MySQL** (misal: users, products, transactions).

- **Alasan:** Sesuai untuk menyimpan data terstruktur, mendukung relasi antar entitas (misalnya transaksi dengan produk dan user), serta mudah di-query dan dijaga konsistensinya.

2. Operasi apa saja yang dibuat terhadap struktur data tersebut?

Operasi CRUD (Create, Read, Update, Delete) telah dibuat, antara lain:

- Tambah pengguna (registerUser)
- Login pengguna (loginUser)
- Tambah produk (addProduct)
- Ambil semua transaksi (getAllTransactions)
- Edit atau hapus data (updateProduct, deleteProduct, dll)

3. Cara akses data agar efisien + contoh program

Untuk efisiensi:

- Menggunakan query yang spesifik seperti `SELECT * FROM products LIMIT 10` (menghindari ambil semua)
- Menghindari nested query
- Validasi input dari frontend sebelum dikirim ke server

Contoh bagian program (controllers/productController.js):

```
js
CopyEdit
const getAllProducts = async (req, res) => {
  try {
    const [products] = await db.query("SELECT * FROM products LIMIT 10");
    res.json(products);
  } catch (err) {
    res.status(500).json({ message: err.message });
  }
};
```

Mengimplementasikan User Interface

4. Simulasi aplikasi yang dibuat + mengacu submodul soal mana?

Aplikasi: **Web toko alat kesehatan**

Fitur:

- Lihat produk
- Tambah ke keranjang
- Checkout transaksi
- Login & registrasi

Submodul: "*Melakukan pembelian secara online*", "*Manajemen produk oleh admin*", dan "*Autentikasi pengguna*".

Melakukan Instalasi Software Tools Pemrograman

5. Pertimbangan memilih tools bahasa pemrograman

- **Node.js + Express:** Ringan dan cepat untuk REST API
 - **React.js:** Komponen UI modular dan efisien untuk frontend
 - **MySQL:** Stabil, terstruktur, dan mendukung relasi antar data
 - Semua tools open source dan kompatibel di berbagai OS
-

Melakukan Kode dengan Prinsip Sesuai Guidelines dan Best Practices

6. Guidelines apa yang diikuti? Contoh bagian program?

Guidelines:

- Pemisahan tanggung jawab (modularisasi: routes, controllers, middlewares)
- Penamaan variabel jelas (userEmail, productName)
- Penanganan error (try-catch di semua controller)

Contoh di `controllers/userController.js`:

```
js
CopyEdit
const registerUser = async (req, res) => {
  const { name, email, password } = req.body;
  try {
    const hashedPassword = await bcrypt.hash(password, 10);
    await db.query("INSERT INTO users (name, email, password) VALUES (?, ?, ?)", [name, email, hashedPassword]);
    res.status(201).json({ message: "User registered successfully" });
  } catch (err) {
    res.status(500).json({ message: err.message });
  }
};
```

7. Kemudahan interaksi pengguna (standard UX/UI)

- Menggunakan **Tailwind CSS** → konsisten dan responsif
- Navigasi jelas: dashboard, login, produk, keranjang
- Form input memiliki label dan placeholder

Contoh form login di frontend:

```
jsx
CopyEdit

```

Menggunakan Library atau Komponen Pre-existing

8. Cara mengidentifikasi class/unit reuse

- Identifikasi dari fitur umum (misal: autentikasi, upload file)
- Cek apakah ada package/library terpercaya di NPM
- Evaluasi dokumentasi dan review pengguna

9. Bagian program dengan ketergantungan antar modul

Contoh di `routes/userRoutes.js`:

```
js
CopyEdit
const express = require('express');
const { loginUser, registerUser } =
require('../controllers/userController');
```

Ketergantungan antara route dan controller untuk pengelolaan user.

Membuat Dokumen Kode Program

10. Bagaimana mengetahui tool dokumentasi sudah sesuai

- Gunakan tools seperti **JSDoc**
 - Hasil dokumentasi dibandingkan dengan struktur fungsi dan komentar di file `.js`
 - Cek apakah semua endpoint dan parameter terdeskripsi dengan baik
-

Melakukan Debugging

11. Cara menelusuri galat/error + contoh

- Gunakan `console.log()` untuk mengecek variabel
- Jalankan dengan `npm run dev` dan pantau log terminal
- Gunakan try-catch untuk melacak error dari DB

Contoh error & cara perbaikan:

Error:

```
sql
CopyEdit
ER_BAD_FIELD_ERROR: Unknown column 'emails' in 'field list'
```

Solusi:

- Kolom seharusnya `email` bukan `emails`

- Periksa dan perbaiki query di file `UserController.js`:

```
js
CopyEdit
await db.query("SELECT * FROM users WHERE email = ?", [email]);
```

Jika kamu ingin saya buat dokumentasi JSDoc atau Swagger dari kode kamu juga, saya bisa bantu! Mau?

1. Sebutkan 2 alternatif konsep dan struktur data yang anda rencanakan dalam pembuatan system anda dan jelaskan kelebihan serta kekurangan masing-masing kedua alternatif tersebut!
2. Pilih sebuah fungsi dari program yang sudah anda buat (fungsi tersebut setidaknya memiliki 1 atau beberapa bagian yang mengandung loop/perulangan). Hitunglah efisiensi memori dan waktu dari fungsi tersebut (catatan : efisiensi memori dinyatakan dalam ukuran memori terpakai dan efisiensi waktu dinyatakan dalam lamanya waktu pemrosesan)
3. Anda telah merancang struktur data yang sangat efisien pada saat perancangan aplikasi. Namun Ketika sampai pada tahap penulisan program, Bahasa pemrograman yang anda gunakan tidak mendukung struktur data tersebut. Apa yang dapat anda lakukan?
4. Sebagai lanjutan kasus pada soal no 3, jika kemudian anda memutuskan untuk mengganti Bahasa pemrograman ke Bahasa lain (yang anda kurang familiar tetapi sangat mendukung struktur data yang telah anda rancang), bagaimana strategi anda untuk tetap dapat menghasilkan aplikasi yang baik dan handal namun juga tetap dapat selesai pada tenggat waktu yang diberikan kepada anda?
5. Pilih sebuah aktivitas dari diagram use case pada soal studi kasus.
 - a. Komponen dialog use interface apa saja yang perlu dibuat dari aktivitas tersebut?
 - b. Tuliskan urutan akses dialog user interface yang dihadapi oleh user dalam menyelesaikan aktivitas pada user case tersebut
6. Pilih sebuah form dari aplikasi yang Anda buat. Bagaimana Anda menentukan bentuk style dari komponen user interface yang ada pada aplikasi Anda agar memenuhi standar aspek desain yang baik? Tunjukkan contohnya dari form yang Anda pilih.
7. Bagaimana cara Anda mengidentifikasi platform (lingkungan) yang sesuai untuk menjalankan tools program pemrograman yang sesuai dengan kebutuhan?
8. Anda telah memilih sebuah versi installer software tools program (sebut saja versi 3.5) untuk diinstal pada komputer Anda. Ternyata versi sistem operasi pada komputer Anda tidak kompatibel dengan versi software tools tersebut. Jika Anda mengganti versi installer software tools dengan versi yang kompatibel dengan sistem operasi (misalnya versi 2.9), Anda akan kehilangan sejumlah fitur kunci dari versi 3.5 yang akan membuat program Anda jauh lebih efisien. Apa yang akan Anda lakukan? Apa pertimbangan Anda?
9. Tuliskan 3 (tiga) coding guidelines yang Anda ketahui!
10. Sebutkan 2 (dua) galat/error yang Anda temui ketika menulis kode. Apa penyebabnya dan bagaimana cara menyelesaikan galat tersebut?
11. Bagaimana cara Anda menghitung efisiensi penggunaan resource dari kode Anda? Berikan contoh sebuah bagian dari kode program Anda yang menunjukkan kode tersebut telah mempertimbangkan faktor efisiensi dan berikan alasannya!

12. Variabel bertipe Integer (int) digunakan untuk menyimpan data Usia. Tepatkah keputusan tersebut jika dikaitkan dengan ukuran memori yang digunakan? Jelaskan alasannya!
13. Dalam pemberian keterangan pada sebuah prosedur/fungsi, apa saja yang perlu dicantumkan? Tunjukkan sebuah prosedur/fungsi yang sudah Anda buat pada program Anda yang telah diberi keterangan.
14. Sebutkan sebuah contoh array yang Anda buat dalam program Anda. Berapa dimensi dari array tersebut? Apa pertimbangan Anda dalam menentukan dimensi array tersebut?
15. Dalam sebuah program, apa perbedaan kesalahan sintaks dengan kesalahan logika? Bagaimana cara Anda mengoreksi kedua macam kesalahan program tersebut?
- 16.
17. Tuliskan sebuah contoh class unit reuse (berasal dari aplikasi lain) yang anda gunakan dalam program anda
18. Dari contoh yang anda tuliskan di soal no.17, seberapa besar efisiensi (dari segi waktu) dari pemanfaatan komponen reuse tersebut terhadap keseluruhan proyek anda
19. Bagaimana cara anda mengidentifikasi ketergantungan antar-unit dalam program anda? Apa yang anda lakukan jika ternyata komponen reuse yang anda gunakan menggunakan komponen/library yang tidak anda miliki?
20. Mengapa pembaharuan library merupakan hal yang penting dalam pembuatan sebuah program? Bagaimana cara Anda melakukan pembaharuan library atau komponen pre-existing pada program Anda?
21. Pilih sebuah fungsi dari program Anda:
 - a. Tuliskan nama class/modul dari fungsi tersebut
 - b. Tuliskan parameter input dan output dari fungsi tersebut
 - c. Jelaskan algoritma dari fungsi tersebut
22. Dari class/modul yang Anda pilih pada soal no. 21, jelaskan secara umum apa kegunaan dari class/modul tersebut terhadap keseluruhan sistem aplikasi yang Anda buat.
23. Jelaskan apa yang dimaksud dengan eksepsi/exception dalam sebuah program. Dari program yang sudah Anda buat, tuliskan bagian yang berpotensi menimbulkan eksepsi dan jelaskan bagaimana cara penanganannya
24. Sebutkan macam-macam software tools untuk men-generate dokumentasi dari bahasa pemrograman yang Anda gunakan.
25. Setelah Anda sukses mengkompilasi kode program Anda, selanjutnya untuk menghasilkan software yang dapat digunakan, Anda perlu melakukan build dari kode program.
 - a. Apakah program Anda lulus build pada percobaan pertama? Jika tidak, kriteria apa saja yang menyebabkan gagal build? Tuliskan sebuah error yang terjadi pada saat percobaan eksekusi aplikasi Anda.
 - b. Faktor apa saja yang menentukan kesuksesan eksekusi sebuah aplikasi?
26. Sebutkan sebuah langkah yang telah Anda lakukan untuk memperbaiki kesalahan saat kompilasi maupun saat build aplikasi.